

VeriGen: A Large Language Model for Verilog Code Generation

Year: 2023 | Citations: 259 | Authors: *Shailja Thakur, Baleegh Ahmad, H. Pearce*

Abstract

In this study, we explore the capability of Large Language Models (LLMs) to automate hardware design by automatically completing partial Verilog code, a common language for designing and modeling digital systems. We fine-tune pre-existing LLMs on Verilog datasets compiled from GitHub and Verilog textbooks. We evaluate the functional correctness of the generated Verilog code using a specially designed test suite, featuring a custom problem set and testing benches. Here, our fine-tuned open-source CodeGen-16B model outperforms the commercial state-of-the-art GPT-3.5-turbo model with a 1.1% overall increase. Upon testing with a more diverse and complex problem set, we find that the fine-tuned model shows competitive performance against state-of-the-art gpt-3.5-turbo, excelling in certain scenarios. Notably, it demonstrates a 41% improvement in generating syntactically correct Verilog code across various problem categories compared to its pre-trained counterpart, highlighting the potential of smaller, in-house LLMs in hardware design automation. We release our training/evaluation scripts and LLM checkpoints as open-source contributions.